Fabric Collective Accelerator (FCA)

User Manual

Version 3.0

www.mellanox.com

NOTE:
THIS HARDWARE, SOFTWARE OR TEST SUITE PRODUCT ("PRODUCT(S)") AND ITS RELATED DOCUMENTATION ARE PROVIDED BY MELLANOX TECHNOLOGIES "AS-IS" WITH ALL FAULTS OF ANY KIND AND SOLELY FOR THE PURPOSE OF AIDING THE CUSTOMER IN TESTING APPLICATIONS THAT USE THE PRODUCTS IN DESIGNATED SOLUTIONS. THE CUSTOMER'S MANUFACTURING TEST ENVIRONMENT HAS NOT MET THE STANDARDS SET BY MELLANOX TECHNOLOGIES TO FULLY QUALIFY THE PRODUCTO(S) AND/OR THE SYSTEM USING IT. THEREFORE, MELLANOX TECHNOLOGIES CANNOT AND DOES NOT GUARANTEE OR WARRANT THAT THE PRODUCTS WILL OPERATE WITH THE HIGHEST QUALITY. ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT ARE DISCLAIMED. IN NO EVENT SHALL MELLANOX BE LIABLE TO CUSTOMER OR ANY THIRD PARTIES FOR ANY DIRECT, INDIRECT, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES OF ANY KIND (INCLUDING, BUT NOT LIMITED TO, PAYMENT FOR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY FROM THE USE OF THE PRODUCT(S) AND RELATED DOCUMENTATION EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Mellanox Technologies
350 Oakmead Parkway Suite 100
Sunnyvale, CA 94085
U.S.A.
www.mellanox.com
Tel: (408) 970-3400
Fax: (408) 970-3403

Mellanox Technologies, Ltd.
Beit Mellanox
PO Box 586 Yokneam 20692
Israel
www.mellanox.com
Tel: +972 (0)74 723 7200
Fax: +972 (0)4 959 3245

Contents

List of Figures

List of Tables

Document Revision History

Revision	Date	Description
Rev 3.0	February 2014	Initial Release

Preface

Audience

The intended audience for the Mellanox Fabric Collective Accelerator (FCA) User Manual is the MPI implementer and the network administrator responsible for managing FCA on Mellanox InfiniBand switches. It is assumed that the administrator is familiar with advanced concepts in network management.

Typographical Conventions

Before you start using this guide, it is important to understand the terms and typographical conventions used in the documentation.

The following kinds of formatting in the text identify special information.

Formatting convention	Type of Information
Special Bold	Items you must select, such as menu options, command buttons, or items in a list.
Emphasis	Use to emphasize the importance of a point or for variable expressions such as parameters.
CAPITALS	Names of keys on the keyboard. for example, SHIFT, CTRL, or ALT.
KEY+KEY	Key combinations for which the user must press and hold down one key and then press another, for example, CTRL+P, or ALT+F4.

Document Conventions

NOTE: Identifies important information that contains helpful suggestions.

CAUTION: Alerts you to risk of personal injury, system damage, or loss of data.

WARNING: Warns you that failure to take or avoid a specific action might result in personal injury or a malfunction of the hardware or software. Be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents before you work on any equipment.

1 Introduction to Mellanox Fabric Collective Accelerator

1.1 Overview

To meet the needs of scientific research and engineering simulations, supercomputers are growing at an unrelenting rate. As supercomputers increase in size from mere thousands to hundreds-of-thousands of processor cores, new performance and scalability challenges have emerged. In the past, performance tuning of parallel applications could be accomplished fairly easily by separately optimizing their algorithms, communication, and computational aspects. However, as systems continue to scale to larger machines, these issues become co-mingled and must be addressed comprehensively.

Collective communications execute global communication operations to couple all processes/nodes in the system and therefore must be executed as quickly and as efficiently as possible. Indeed, the scalability of most scientific and engineering applications is bound by the scalability and performance of the collective routines employed. Most current implementations of collective operations will suffer from the effects of systems noise at extreme-scale (system noise increases the latency of collective operations by amplifying the effect of small, randomly occurring OS interrupts during collective progression.) Furthermore, collective operations will consume a significant fraction of CPU cycles, cycles that could be better spent doing meaningful computation.

Mellanox Technologies has addressed these two issues, lost CPU cycles and performance lost to the effects of system noise, by offloading the communications to the host channel adapters (HCAs) and switches. The technology, named CORE-Direct® (Collectives Offload Resource Engine), provides the most advanced solution available for handling collective operations thereby ensuring maximal scalability, minimal CPU overhead, and providing the capability to overlap communication operations with computation allowing applications to maximize asynchronous communication.

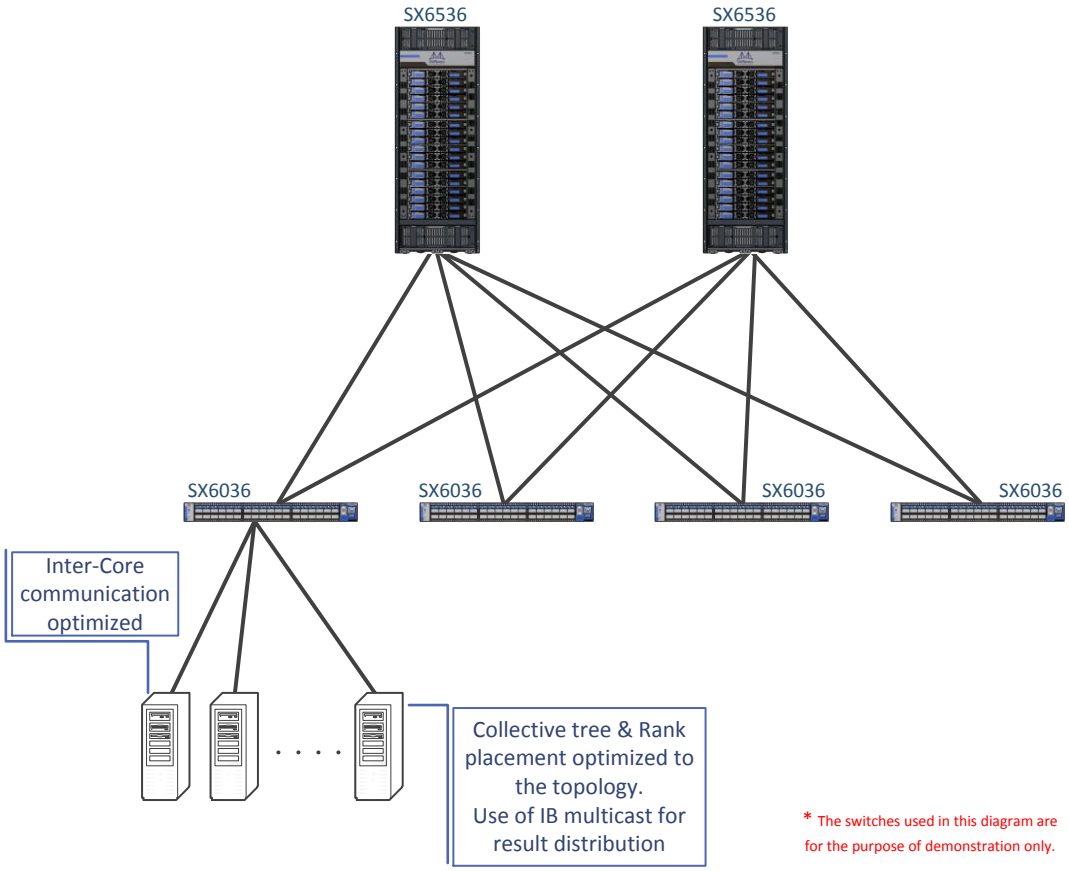
Users may benefit immediately from CORE-Direct® out-of-the-box by simply specifying the necessary BCOL/SBGP combinations. In order to take maximum advantage of CORE-Direct®, users may modify their applications to use MPI 3.0 non-blocking routines while using CORE-Direct® to offload the collective "under-the-covers", thereby allowing maximum opportunity to overlap communication with computation.

Additionally, FCA 3.0 also contains support to build runtime configurable hierarchical collectives. We currently support socket and UMA level discovery with network topology slated for future versions. As with FCA 2.X we also provide the ability to accelerate collectives with hardware multicast. In FCA 3.0 we also expose the performance and scalability of Mellanox's advanced point-to-point library, MXM 2.x, in the form of the "mlnx_p2p" BCOL. This allows users to take full advantage of new features with minimal effort.

FCA 3.0 is a standalone library that can be integrated into any MPI or PGAS runtime. Support for FCA 3.0 is currently integrated into Open MPI versions 1.7.4 and higher. The 3.0 release currently supports blocking and non-blocking variants of “Allgather”, “Allreduce”, “Barrier”, and “Bcast”.

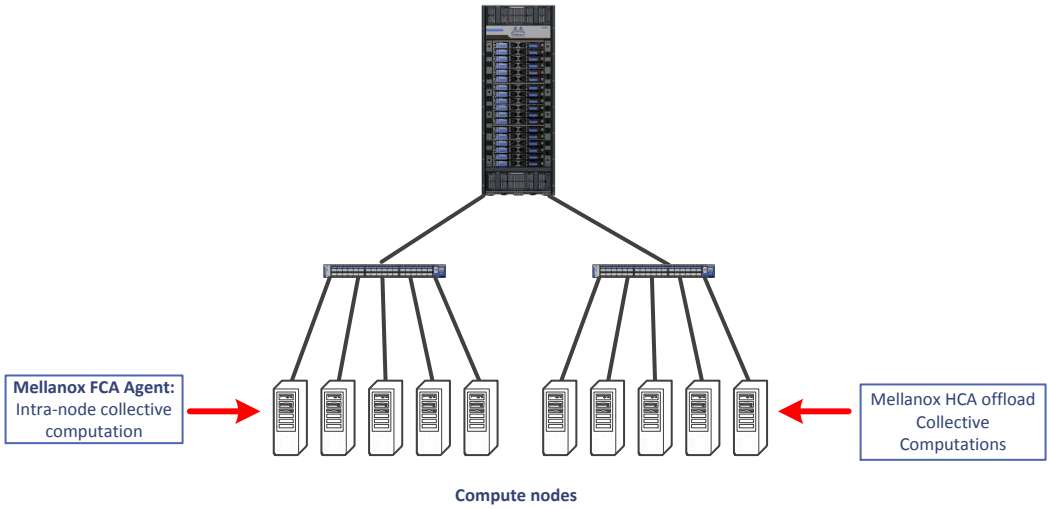
The following diagram summarizes the FCA architecture:

Figure 1: FCA Architecture



The following diagram shows the FCA components and the role that each plays in the acceleration process:

Figure 2: FCA Components



1.2 FCA Installation Package Content

The FCA installation package includes the following items:

- FCA - Mellanox Fabric Collector Accelerator Installation files
  - hcoll-<version>.x86_64.<OS>.rpm
  - hcoll-<version>.x86_64.<OS>.tar.gz

    where:

    <version> is the version of this release, and

    <OS> is one of the supported Linux distributions listed in Prerequisites (on page 11).
- Mellanox Fabric Collective Accelerator (FCA) Software: End-User License Agreement
- FCA MPI runtime libraries
- Mellanox Fabric Collective Accelerator (FCA) User Manual
- Mellanox Fabric Collective Accelerator (FCA) Release Notes

1.3 Differences Between FCA v2.5 and FCA v3.0

FCA v3.0 is new software which exposes the power of CORE-Direct® to offload collective operations to the HCA. It adds additional scalable algorithms for collectives and supports both blocking and non-blocking APIs (MPI-3 SPEC compliant).

2 Installation and Initial Configuration

2.1 Downloading the FCA Software

This software download process applies to software updates as well as for first time installation.

➢ ***To download the FCA software***

1. Go to the Mellanox website.
2. Click the **Downloads** tab and select the relevant version of the software to download.
3. Save the file on your local drive.
4. Click **Close**.

2.1.1 Downloading the MXM Software

➢ ***To download the MXM software***

1. Go to the Mellanox website.
2. Click the **Downloads** tab and select the relevant version of the software to download.
3. Save the file on your local drive.
4. Click **Close**.

2.2 Installing FCA

2.2.1 Prerequisites

Before you begin be certain that:

1. Mellanox OFED 2.1-1.0.0 or later is installed.

   To download the latest MLNX_OFED version, go to:

   Mellanox OpenFabrics Enterprise Distribution for Linux (MLNX_OFED)

2. Mellanox ConnectX®-3 / ConnectX®-3 Pro and Connect-IB® HCA.

   To download the latest ConnectX® family / Connect-IB® HCA firmware version, go to:

   Firmware Downloads

The minimum system requirements for installing and running FCA are listed in the following table.

Table 1: System Requirements

Item	Requirement
FCA 3.0	
Supported switches	Mellanox IB QDR/FDR switches
Linux distributions (<OS>)	• RHEL/CentOS 6.3, 6.4, 6.5 • SLES11 SP1, SLES11 SP2, SLES11 SP3

Item	Requirement
	• OEL 6.1, 6.2, 6.3, 6.4 • Citrix XenServer Host 6.x • Fedora 18, 19 • Ubuntu 12.04, 13.04, 13.10 • Debian 6.0.7, 6.0.8, 7.1, 7.2 • kernel 3.10, 3.11, 3.12
Supported HCAs	Mellanox ConnectX®-3 / ConnectX®-3 Pro HCA with firmware version 2.30.8000 or later Mellanox Connect-IB® HCA with firmware version 10.10.2000 or later
Open Message Passing Interface (MPI) Project	Open MPI 1.7.4 or later.*
MLNX_OFED	2.1-1.0.0 or later
Root permission	The installer should have root permissions for post-installation tasks.
InfiniBand Subnet Management	All InfiniBand Subnet Management based software is supported in FCA version 3.0.

3 Configuring FCA

3.1 Compiling Open MPI with FCA 3.0

➢ ***To compile Open MPI with FCA 3.0***

1. Install FCA 3.0 from:
   - an RPM.

     ```
     % rpm -ihv hcoll-x.y.z-1.x86_64.rpm
     ```
   - a tarball.

     ```
     % tar jxf hcoll-x.y.z.tbz
     ```

   FCA 3.0 will be installed automatically in the /opt/mellanox/hcoll folder.
2. Enter the Open MPI source directory and run the following command:

```
% cd $OMPI_HOME
% ./configure --with-hcoll=/opt/mellanox/hcoll --with-mxm=/opt/mellanox/mxm
< ... other configure  parameters>
% make -j 9 && make install -j 9
```

NOTE: libhcoll required MXM v2.1 or higher.

➢ ***To check the version of FCA installed on your host***

```
% rpm -qi hcoll
```

➢ ***To upgrade to a newer version of FCA 3.X***

1. Remove the existing FCA 3.X.

   ```
   %rpm -e hcoll
   ```
2. Remove the precompiled Open MPI.

   ```
   %rpm -e mlnx-openmpi_gcc
   ```
3. Install the new FCA 3.X and compile the Open MPI with it.

3.2 Enabling FCA in Open MPI

To enable the use of the FCA 3.0 HCOLL collectives in Open MPI, you need to explicitly ask for them by setting the following MCA parameter:

```
mpirun -np 32 --display-map --bind-to-core -mca coll hcoll,tuned,libnbc,basic
-mca btl_openib_if_include mlx4_0:1 -mca coll_hcoll_np 0 -x
HCOLL_MAIN_IB=mlx4_0:1 -x HCOLL_BCOL=basesmuma,mlnx_p2p -x
HCOLL_SBGP=basesmuma,p2p ./a.out
```

3.3 Tuning FCA 3.0 Setting

The default FCA 3.0 settings should be optimal for most systems. However, to check the available FCA 3.0 parameters and their default values, run the following command:

```
%%/opt/mellanox/hcoll/bin/hcoll_info --all
```

FCA 3.0 parameters are simply environment variables and can be modified in one of two ways:

- Modify the default FCA 3.0 parameters as part of the mpirun command

```
%mpirun ... -x HCOLL_ML_BUFFER_SIZE=65536
```

- Modify the default FCA 3.0 parameter values from SHELL:

```
% export -x HCOLL_ML_BUFFER_SIZE=65536
% mpirun ...
```

3.4 Selecting Ports and Devices

Select which HCA device and port you would like FCA 3.0 to run over by setting:

```
-x HCOLL_MAIN_IB=<device_name>:<port_num>
```

4 Runtime Configuration of FCA

4.1 Memory Hierarchy

FCA 3.0 is flexible and modular, providing the user a wide degree of latitude to customize collective algorithms to take full advantage of their Mellanox hardware at application runtime.

The FCA 3.0 software model abstracts the notion of a memory hierarchy into subgrouping or SBGP components. An SBGP group is a subset of endpoints that satisfy a reachability criterion, for example, all processes on the same socket. To each SBGP is associated a set of optimized collective primitives, basic collectives or BCOL components.

4.1.1 Available SBGPs

Table 2: Available SBGPs

SBGPs	Description
basesmuma	A subset of ranks that share the same host.
basesmsocket	A subset of ranks that share the same socket.
ibnet	A subset of ranks that can communicate with CORE-Direct®.
p2p	A subset of ranks that can reach each other over point-to-point.

4.1.2 Available BCOLs

Table 3: Available BCOLs

BCOLs	Description
basesmuma	Shared memory collective primitives.
mlnx_p2p	MXM based point-to-point collective primitives.
iboffload	CORE-Direct® based collective primitives.
ptpcoll	Point-to-point logical layer.

4.1.3 Supported Collectives

Table 4: Supported Collectives

Collectives	Description
Allgather/Iallgather	Blocking and non-blocking allgather for all possible bcol/sbgp combinations
Allreduce/Iallreduce	Blocking and non-blocking allreduce. **Note:** Currently not supported with iboffload BCOL.
Barrier/Ibarrier	Blocking and non-blocking barrier for all possible BCOL/SBGP combinations.
Bcast/Ibcast	Blocking and non-blocking bcast for all possible BCOL/SBGP combinations.

4.1.4 Different Memory Hierarchy Usages

- Two-level hierarchy with CORE-Direct® used at the "top" level:

```
% mpirun -x HCOLL_BCOL=basesmuma,iboffload,mlnx_p2p -x
HCOLL_SBGP=basesmuma,ibnet,p2p
```

- Three-level hierarchy with CORE-Direct® used at the "top" level:

```
% mpirun -x HCOLL_BCOL=basesmuma,basesmuma,iboffload,mlnx_p2p -x
HCOLL_SBGP=basesmsocket,basesmuma,ibnet,p2p
```

- Two-level hierarchy with MXM p2p used at the "top" level:

```
% mpirun -x HCOLL_BCOL=basesmuma,mlnx_p2p -x HCOLL_SBGP=basesmuma,p2p
```

- Three-level hierarchy with MXM used at the "top" level:

```
% mpirun -x HCOLL_BCOL=basesmuma,basesmuma,mlnx_p2p -x
HCOLL_SBGP=basesmsocket,basesmuma,p2p
```

4.2 Enabling Mellanox Specific Features and Optimizations

- Multicast acceleration:

  FCA 3.0 uses hardware multicast to accelerate collective primitives in both the "`mlnx_p2p`" and "`iboffload`" BCOLs when possible.

  To enable multicast based collectives, set:

```
-x HCOLL_MCAST_ENABLE_ALL=1
```

- Context caching:

  When using one of the two Mellanox specific BCOLs (`mlnx_p2p`, or `iboffload`), you may enable context caching. This optimization can benefit applications that create and destroy many MPI communicators.

  To enable context caching in conjunction with a valid BCOL/SBGP pair, set:

```
-x HCOLL_CONTEXT_CACHE_ENABLE=1
```

4.3 Selecting Shared Memory and MXM Point-To-Point Hierarchies for Collectives in FCA

Running IMB benchmark on 1,024 MPI processes with two levels of hierarchy:

- shared memory
- MXM point-to-point

Enable both context caching and multicast acceleration.

```
% mpirun -np 1024 --bind-to-core -bynode -mca btl_openib_if_include mlx4_0:1
-mca coll hcoll,tuned,libnbc -mca btl sm,openib,self HCOLL_MCAST_ENABLE_ALL=1
-x HCOLL_ENABLE_CONTEXT_CACHE=1
-x HCOLL_IB_IF_INCLUDE=mlx4_0:1 -x HCOLL_BCOL=basesmuma,mlnx_p2p
-x HCOLL_SBGP=basesmuma,p2p ~/IMB/src/IMB-MPI1 -exclude PingPong PingPing
Sendrecv
```

5 FCA 3.0 Integration

In principle, FCA 3.0 can be integrated into any communication library. In order to do so, one must first implement the so-called “RTE interface”, which is a collection of callbacks and handles that must be supplied to FCA 3.x from the calling library. For an example of full integration into an MPI library, please refer to the Open MPI source code under ompi_src/ompi/mca/coll/hcoll.

The "hcoll" component contained in the OMPI "coll" framework is the runtime integration layer of FCA into OMPI. A complete implementation of the RTE can be found at ompi_src/ompi/mca/coll/hcoll. A standalone example can be found at `/opt/mellanox/hcoll/sdk`.

Please refer to the SDK's README for instructions on compiling and running. The RTE implementation can be found in the "hcoll_sdk.c" file.